JOURNAL OF
COMPUTATIONAL AND
APPLIED MATHEMATICS

ELSEVIER Journal of Computational and Applied Mathematics 103 (1999) 301–306

A computable error bound for matrix functionals

D. Calvetti[a,1], G.H. Golub[b,2], L. Reichel[c,*,3]

[a] *Department of Mathematics, Case Western Reserve University, Cleveland, OH 44106, U.S.A.*

[b] *Computer Science Department, Stanford University, Stanford, CA 93405, U.S.A.*

[c] *Department of Mathematics and Computer Science, Kent State University, Kent, OH 44242, U.S.A.*

Received 20 September 1998; received in revised form 29 October 1998

Abstract

Many problems in applied mathematics require the evaluation of matrix functionals of the form $F(A):=u^{\mathrm{T}}f(A)u$, where A is a large symmetric matrix and u is a vector. Golub and collaborators have described how approximations of such functionals can be computed inexpensively by using the Lanczos algorithm. The present note shows that error bounds for these approximations can be computed essentially for free when bounds for derivatives of f on an interval containing the spectrum of A are available.

1. Introduction

The evaluation of matrix functionals of the form

$$F(A):=u^{\mathrm{T}}f(A)u, \quad A\in\mathbb{R}^{n\times n}, \; u\in\mathbb{R}^{n}, \tag{1.1}$$

where A is a large, possibly sparse, symmetric matrix, and u is a vector, arises in many applications. The matrix $f(A)$ is defined in terms of the spectral factorization of A, i.e.,

$$f(A):=Sf(\Lambda)S^{\mathrm{T}}, \tag{1.2}$$

* Corresponding author. E-mail: na.reichel@na.net.ornl.gov.

[1] Research supported in part by NSF grant DMS-9896073.

[2] Research supported in part by NSF grant CCR-9505393.

[3] Research supported in part by NSF grant DMS-9404706.

0377-0427/99/\$ – see front matter
PII: S0377-0427(98)00267-2

where

$$A = S\Lambda S^{\mathrm{T}}, \qquad \Lambda = \mathrm{diag}[\lambda_1, \lambda_2, \ldots, \lambda_n], \qquad S^{\mathrm{T}} S = I_n.$$

Here and elsewhere in this paper I_k denotes the $k \times k$ identity matrix. For notational simplicity, we will assume that the eigenvalues are ordered so that

$$\lambda_1 \leqslant \lambda_2 \leqslant \cdots \leqslant \lambda_n,$$

and that the vector u in (1.1) satisfies $\|u\| = 1$, where $\|\cdot\|$ denotes the Euclidean vector norm.

Golub and collaborators have, in a sequence of papers, described how approximations of $F(A)$ can be evaluated inexpensively by exploiting the connection between matrix functionals of the form (1.1), Stieltjes integrals, Gauss-type quadrature rules and the Lanczos algorithm; see, e.g., [1–3, 5] and references therein.

For definiteness, introduce the vector $[\mu_1, \mu_2, \ldots, \mu_n] := u^{\mathrm{T}} S$ and, using (1.2), write the functional (1.1) as

$$F(A) = u^{\mathrm{T}} S f(\Lambda) S^{\mathrm{T}} u = \sum_{j=1}^{n} f(\lambda_j) \mu_j^2. \tag{1.3}$$

The right-hand side of (1.3) is a Stieltjes integral

$$\mathscr{I} f := \int_{-\infty}^{\infty} f(t)\, \mathrm{d}\mu(t)$$

with a nonnegative measure $\mathrm{d}\mu(t)$, such that $\mu(t)$ is a nondecreasing step function defined on $\mathbb{R}$ with jumps of height μ_j^2 at $t = \lambda_j$. It follows from $\|u\| = 1$ that the measure $\mathrm{d}\mu(t)$ is of total mass one. The m-point Gauss quadrature rule associated with the measure $\mathrm{d}\mu(t)$ is of the form

$$\mathscr{G}_m f := \sum_{j=1}^{m} f(\theta_j) \gamma_j^2 \tag{1.4}$$

and is characterized by the fact that

$$\mathscr{I} f = \mathscr{G}_m f, \quad \forall f \in \mathbf{P}_{2m-1},$$

where $\mathbf{P}_{2m-1}$ denotes the set of polynomials of degree at most $2m-1$. The nodes θ_j of the quadrature rule are known to be the zeros of an mth degree orthogonal polynomial with respect to the inner product

$$(f, g) := \mathscr{I}(fg). \tag{1.5}$$

Assume that the nodes are ordered according to $\theta_1 < \theta_2 < \cdots < \theta_m$. It is well known that for a $2m$ times continuously differentiable function f, the error in the quadrature rule (1.4) can be expressed as

$$\mathscr{E}_m f := (\mathscr{I} - \mathscr{G}_m) f = \frac{f^{(2m)}(\theta_{\mathscr{G}})}{(2m)!} \cdot \int_{-\infty}^{\infty} \prod_{j=1}^{m} (t - \theta_j)^2 \mathrm{d}\mu(t) \tag{1.6}$$

for some $\theta_{\mathscr{G}} \in [\lambda_1, \lambda_n]$. If the derivative $f^{(2m)}$ does not change sign in $[\lambda_1, \lambda_n]$, then the sign of the error $\mathscr{E}_m f$ can be determined from (1.6). For instance, if $f^{(2m)}(t) > 0$ for $\lambda_1 \leqslant t \leqslant \lambda_n$, then $\mathscr{E}_m f > 0$ and therefore $\mathscr{G}_m f < \mathscr{I} f$. When in addition $f^{(2m+1)}(t)$ does not change sign in the interval $[\lambda_1, \lambda_n]$, an upper bound for $\mathscr{I} f$ can be determined analogously by using an $(m+1)$-point Gauss–Radau quadrature rule (see [3]). However, when $f^{(2m)}$ or $f^{(2m+1)}$ change sign in $[\lambda_1, \lambda_n]$, then Gauss and Gauss–Radau rules might not yield lower and upper bounds for $\mathscr{I} f$.

The present note shows that the quantities required to evaluate the approximation $\mathscr{G}_m f$ allow the computation of a bound for the right-hand side of (1.6) essentially for free when a bound for $|f^{(2m)}|$ on the interval $[\lambda_1, \lambda_n]$ is available. This bound does not require $f^{(2m)}$ to be of one sign. Details are described in Section 2. A numerical example is presented in Section 3.

2. An error bound

We first discuss the computation of $\mathscr{G}_m f$, and then show how a bound for the error $\mathscr{E}_m f$ can be determined with very little additional work. The computation of Gauss quadrature rules is discussed in [1–3, 5]. Our review of these results allows us to introduce notation necessary to discuss the evaluation of the error bound.

Gauss quadrature rules with respect to the measure $d\mu(t)$ can conveniently be determined by the Lanczos algorithm. Application of m steps of the Lanczos algorithm to the matrix A with initial vector $v_1 := u$ yields the decomposition

$$AV_m = V_m T_m + \beta_m v_{m+1} e_m^{\mathrm{T}}, \tag{2.1}$$

where $V_m = [v_1, v_2, \ldots, v_m] \in \mathbb{R}^{n \times m}$ and $v_{m+1} \in \mathbb{R}^n$ satisfy $V_m^{\mathrm{T}} V_m = I_m$, $v_{m+1}^{\mathrm{T}} v_{m+1} = 1$, $V_m^{\mathrm{T}} v_{m+1} = 0$, $\beta_m \in \mathbb{R}$, and e_j denotes the jth axis vector. Moreover,

$$T_m := \begin{bmatrix} \alpha_1 & \beta_1 & & & & \mathbf{0} \\ \beta_1 & \alpha_2 & \beta_2 & & & \\ & \beta_2 & \alpha_3 & & & \\ & & & \ddots & & \\ & & \ddots & \ddots & & \beta_{m-1} \\ \mathbf{0} & & & & \beta_{m-1} & \alpha_m \end{bmatrix} \in \mathbb{R}^{m \times m}$$

is a symmetric tridiagonal matrix. We refer to [4, Ch. 9] for a detailed discussion of the Lanczos algorithm.

The relation (2.1) between the columns v_j of V_m shows that

$$v_j = p_{j-1}(A) v_1, \quad 1 \leqslant j \leqslant m+1, \tag{2.2}$$

for certain polynomials p_{j-1} of degree $j-1$. It follows from the orthonormality of the vectors v_j that

$$
\begin{aligned}
(p_{j-1}, p_{k-1}) &= \int_{-\infty}^{\infty} p_{j-1}(t) p_{k-1}(t)\, \mathrm{d}\mu(t) = u^{\mathrm{T}} S p_{j-1}(\Lambda) p_{k-1}(\Lambda) S^{\mathrm{T}} u \\
&= u^{\mathrm{T}} p_{j-1}(A) p_{k-1}(A) u = v_1^{\mathrm{T}} p_{j-1}(A) p_{k-1}(A) v_1 \\
&= v_j^{\mathrm{T}} v_k = \begin{cases} 0, & j \neq k, \\ 1, & j = k. \end{cases}
\end{aligned} \tag{2.3}
$$

Thus, the polynomials p_j are orthogonal with respect to the inner product (1.5). Combining (2.1) and (2.2) yields a recurrence relation for the polynomials p_j:

$$
\begin{aligned}
&\beta_1 p_1(t) = (t - \alpha_1) p_0(t), \quad p_0(t) = 1, \\
&\beta_j p_j(t) = (t - \alpha_j) p_{j-1}(t) - \beta_{j-1} p_{j-2}(t), \quad 2 \leqslant j \leqslant m,
\end{aligned} \tag{2.4}
$$

which can be expressed as

$$
[p_0(t), p_1(t), \ldots, p_{m-1}(t)] T_m = t[p_0(t), p_1(t), \ldots, p_{m-1}(t)] - \beta_m [0, \ldots, 0, p_m(t)]. \tag{2.5}
$$

In particular, Eq. (2.5) shows that the zeros of p_m are the eigenvalues of T_m.

Introduce the spectral decomposition

$$
T_m = Q_m D_m Q_m^{\mathrm{T}}, \qquad D_m = \mathrm{diag}[\theta_1, \theta_2, \ldots, \theta_m], \qquad Q_m^{\mathrm{T}} Q_m = I_m.
$$

It is well known that the weights of the Gauss rule (1.4) are given by $\gamma_j^2 = (e_1^{\mathrm{T}} Q_m e_j)^2$, $1 \leqslant j \leqslant m$. It follows that the Gauss rule can be written in the form

$$
\mathscr{G}_m f = e_1^{\mathrm{T}} Q_m f(D_m) Q_m^{\mathrm{T}} e_1 = e_1^{\mathrm{T}} f(T_m) e_1. \tag{2.6}
$$

Hence, $\mathscr{G}_m f$ can be determined by first computing the Lanczos decomposition (2.1) and then evaluating one of the expressions (2.6).

We turn to the evaluation of a bound for the error (1.6). Introduce the monic orthogonal polynomials $q_j \in \mathbf{P}_j$ with respect to the inner product (1.5). It follows from (2.4) that

$$
\begin{aligned}
q_0(t) &:= 1, \\
q_j(t) &:= \beta_j \beta_{j-1} \cdots \beta_1 p_j(t), \quad j \geqslant 1.
\end{aligned} \tag{2.7}
$$

Note that the integral on the right-hand side of (1.6) can be written as

$$
\int_{-\infty}^{\infty} \prod_{j=1}^{m} (t - \theta_j)^2 \, \mathrm{d}\mu(t) = (q_m, q_m). \tag{2.8}
$$

Proposition 2.1. *Assume that there is a constant C_m, such that*

$$
|f^{(2m)}(t)| \leqslant C_m, \quad \lambda_1 \leqslant t \leqslant \lambda_n.
$$

Table 3.1
$F(A) = u^{\mathrm{T}} \sin(A)u$, A symmetric indefinite

$n = 300$	$m = 3$	$m = 4$
$F(A)$	$1.208409205 \times 10^{-1}$	$1.208409205 \times 10^{-1}$
$\mathscr{G}_m f$	$1.208418800 \times 10^{-1}$	$1.208409177 \times 10^{-1}$
$\mathscr{B}_m f$	3.1×10^{-5}	1.3×10^{-7}
$\mathscr{E}_m f$	-9.6×10^{-7}	2.8×10^{-9}

Then

$$|\mathscr{E}_m f| \leqslant \frac{C_m}{2m!} \beta_m^2 \beta_{m-1}^2 \cdots \beta_1^2. \tag{2.9}$$

Proof. The result follows by substituting (2.8) into (1.6), and the observation that

$$(q_m, q_m) = \beta_m^2 \beta_{m-1}^2 \cdots \beta_1^2. \tag{2.10}$$

Eq. (2.10) is a consequence of (2.7) and (2.3). □

In order to evaluate the Gauss rule $\mathscr{G}_m f$ by (2.6), we first applied m steps of the Lanczos algorithm to determine the decomposition (2.1). This gives the coefficients $\beta_1, \beta_2, \ldots, \beta_m$ required in the error bound (2.9). Thus, the error bound (2.9) can be computed with very little arithmetic work in addition to the computations necessary to evaluate the Gauss rule $\mathscr{G}_m f$.

3. A computed example

The numerical example of this section illustrates the bound (2.9). The computations were carried out in MATLAB 5.01 on a MICRON personal computer, i.e., with approximately 15 significant digits.

Example 3.1. We would like to determine an approximation of

$$F(A) := u^{\mathrm{T}} \sin(A)u,$$

where A is a 300×300 real symmetric matrix with randomly generated uniformly distributed eigenvalues in the interval $[-1, 1]$, and u is a random vector of unit length. Thus, $f(t) = \sin(t)$ and the derivatives of f of even order change sign in any interval containing the spectrum of A. This implies that the evaluation of pairs of Gauss and Gauss–Radau quadrature rules as described in [3] is not guaranteed to yield upper and lower bounds for the error $\mathscr{E}_m f$. Table 1 shows the exact value of $F(A)$, and the approximations $G_m f$ for $m = 3$ and 4 computed as described in Section 2. The derivatives of f are bounded by one in magnitude, and therefore, by (2.9),

$$|\mathscr{E}_m f| \leqslant \mathscr{B}_m f := \frac{1}{2m!} \beta_m^2 \beta_{m-1}^2 \cdots \beta_1^2.$$

Table 1 displays the error bound $\mathscr{B}_m f$ as well as $\mathscr{E}_m f$.

We remark that the bound (2.9) is primarily of interest for matrix functionals $F(A)$ defined by integrands f with derivatives that change sign on an interval containing the spectrum of A. When the derivatives of f do not change sign, then pairs of Gauss and Gauss–Radau quadrature rules can be applied to determine upper and lower bounds for $\mathscr{E}_m f$ as described in [3], and in our experience these bounds are often sharper than the bound (2.9).

References

[1] Z. Bai, G.H. Golub, Some large scale matrix computation problems, J. Comput. Appl. Math. 74 (1996) 71–89.
[2] D. Calvetti, G.H. Golub, L. Reichel, Estimation of the L-curve via Lanczos bidiagonalization, BIT, to appear.
[3] G.H. Golub, G. Meurant, Matrices, moments and quadrature, in: D.F. Griffiths, G.A. Watson (Eds.), Numerical Analysis 1993, Longman, Essex, England, 1994, pp. 105–156.
[4] G.H. Golub, C.F. Van Loan, Matrix Computations, 3rd ed., Johns Hopkins University Press, Baltimore, 1996.
[5] G.H. Golub, U. von Matt, Generalized cross-validation for large scale problems, J. Comput. Graph. Stat. 6 (1997), 1–34.